#include<stdio.h>

#include<stdlib.h>

struct node{

int data;

struct node \*link;

};

typedef struct node \*nodeptr;

nodeptr avail;

void createanl()

{

avail->link=NULL;

}

nodeptr getnode()

{

nodeptr node1;

if(avail)

{

node1=avail;

avail=avail->link;

}

else

{

node1=(struct node\*)malloc(sizeof(struct node));

}

return node1;

}

void attachtoempty(int data,nodeptr \*ptr)

{

nodeptr temp;

temp=getnode();

temp->link=(\*ptr);

(\*ptr)->link=temp;

temp->data=data;

\*ptr=temp;

}

void attachtofront(int data,nodeptr \*ptr){

nodeptr temp;

temp=getnode();

if((\*ptr)->link==NULL)

{

attachtoempty(data,ptr);

}

else

{

temp->data=data;

temp->link=(\*ptr)->link;

(\*ptr)->link=temp;

(\*ptr)=temp;

}

}

void attachafter(int data,int item,nodeptr \*ptr)

{

if((\*ptr)->link==NULL)

{

printf("No element in the list");

}

else{

nodeptr node1,x;

x=\*ptr;

(\*ptr)=(\*ptr)->link;

do

{

if ((\*ptr)->data==item)

{

node1=getnode();

node1->data=data;

node1->link=(\*ptr)->link;

(\*ptr)->link=node1;

}

(\*ptr)=(\*ptr)->link;

} while (\*ptr!=x);

}

}

void deleteNode(nodeptr \*ptr, int key) {

if ((\*ptr)->data == key && (\*ptr)->link == \*ptr) {

free(\*ptr);

}

nodeptr temp = \*ptr, d;

if ((\*ptr)->data == key) {

while (temp->link != (\*ptr)) temp = temp->link;

temp->link = (\*ptr)->link;

free(\*ptr);

\*ptr = temp;

}

while (temp->link != \*ptr && temp->link->data != key) {

temp = temp->link;

}

if (temp->link->data == key) {

d = temp->link;

temp->link = d->link;

free(d);

}

}

void display(nodeptr ptr)

{

(ptr)=(ptr)->link;

nodeptr starta=(ptr);

(ptr)=(ptr)->link;

for (;ptr!=starta;(ptr)=(ptr)->link)

{

printf("%d\t",(ptr)->data);

}

}

void main()

{

nodeptr lasta;

lasta=getnode();

lasta->data=0;

lasta->link=NULL;

int b=1,data,item,key,k;

while(b!=0)

{

printf("enter 1 to insert in front of the list\n enter 2 to insert after an element\n enter 3 to delete an element\n enter 4 to display\n");

scanf("%d",&k);

switch (k)

{

case 1:

printf("enter the data:\n");

scanf("%d",&data);

attachtofront(data,&lasta);

break;

case 2:

printf("Enter the element after which to be inserted\n");

scanf("%d",&item);

printf("enter the data:\n");

scanf("%d",&data);

attachafter(data,item,&lasta);

break;

case 3:

printf("Enter the element to be deleted\n");

scanf("%d",&key);

deleteNode(&lasta,key);

break;

case 4:

display(lasta);

}

printf("Enter 0 to exit:\n");

scanf("%d",&b);

}

}